



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/804,889	03/19/2004	Bryan Dumm	BCP-001	8180
7590 William L. Botjer PO Box 478 Center Moriches, NY 11934			EXAMINER ZHEN, L I B	
			ART UNIT 2194	PAPER NUMBER
			MAIL DATE 05/13/2008	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/804,889

Applicant(s)

DUMM, BRYAN

Examiner

Li B. Zhen

Art Unit

2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 19 March 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-20 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-20 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO/SF/ICE)
- Paper No(s)/Mail Date 3/19/2007
- 4) ☐ Interview Summary (PTO-413)
- Paper No(s)/Mail Date _____
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____

DETAILED ACTION

1. Claims 1 – 20 are pending in the application.

Claim Objections

2. Claims 15 and 19 are objected to because of the following informalities:
 - a. Claim 15, line 30 (step f, ii) should end with a semicolon;
 - b. Claim 19, line 29 (step e, i, c) should end with a semicolon instead of a period;
 - c. Appropriate correction is required.

Specification

3. The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required: the specification fails to provide antecedent basis for the term “computer usable medium”.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 10 – 14 and 19 – 20 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claims 10 and 19 recite systems comprising of only software. It is unclear if it is applicant's intention to claim a system include computer hardware or a system of

Art Unit: 2194

software only. For the purpose of examination, examiner will interpret the system as a software system.

Claim Rejections - 35 USC § 101

6. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

7. Claims 10 – 20 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 10 and 19 recite a system comprising profile generator, network map generator, administrative list generator, front-end, back-end, event-driven machine and controllers, which are software. Currently presented claims 10 and 19 do not include any computer hardware; therefore, the systems as recited in claim 10 and 19 are considered to include software only. Computer software is functional descriptive material; however, function descriptive material is nonstatutory when claimed as descriptive material per se. When functional descriptive material is recorded on some computer-readable medium, it becomes structurally and functionally interrelated to the medium and will be statutory in most cases since use of technology permits the function of the descriptive material to be realized. Since claims 10 and 19 do not recite the software as being recorded on a computer-readable medium, the system is interpreted as comprising functional descriptive material per se and non statutory. See MPEP § 2106.01.

Claim 15 recites a "computer usable medium" and the specification fails to provide antecedent bases for this limitation [see objection to the specification above]. Without antecedent basis for "computer usable medium", it is unclear if the limitation intended to be the same as the storage media described as part of the disclosed program product or whether it's intended to be broader than the disclosed storage media. It is believed that the limitation "computer usable medium" is intended to claim something broader than the disclosed storage media and cover signals, waves and other forms of transmission media, that carry instructions. Therefore, the limitation "computer usable medium" is not limited to physical articles or objects which constitute a manufacture within the meaning of 35 USC 101 and enable any functionality of the instructions carried thereby to act as a computer component and realize their functionality. As such, the claim is not limited to statutory subject matter and is therefore non-statutory.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

9. **Claims 1 – 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent Application Publication No. 20050138557 A1 to Bolder et al.**

**[hereinafter Bolder] in view of U.S. Patent No. 7,113,989 to Murray et al.
[hereinafter Murray].**

10. As to claim 1, Bolder teaches a method for capturing administrative processes in a machine-readable format [script management system 100 processes parameterized script templates into command scripts; p. 3, paragraph 0039], the administrative processes being processes that need to be implemented by an administrator for achieving different objectives in a local or a networked environment [p. 3, paragraph 0041], the administrative processes comprising at least one administrative task and the administrative task comprising administrative commands [p. 4, paragraph 0059], the method comprising the steps of:

b. generating profiles by capturing the administrative commands in the generic command framework in a machine-readable format [a script management system 100 for performing operations on parameterized CLI script templates 122 to generate managed entity target specific scripts; p. 3, paragraph 0037], the profiles being collections of machine-readable administrative commands that define an administrative task [custom CLI command script for each specific target managed entity 232; p. 4, paragraph 0059];

c. generating network maps, each network map comprising details of one or more servers in the networked environment [NMS 200 provides 220 network maps generated from managed entity instances 216 registered with the NMS 200 and. relationships 214 therebetween; p. 3, paragraph 0050];

d. generating administrative lists, each administrative list comprising one or more profiles that define administrative tasks, which constitute an administrative process [list of script templates 122 selected for managed entity configuration, is hereinafter referred to as an "apply list"; pp. 3 – 4, paragraph 0051];

wherein each administrative list captures an administrative process in a machine-readable format, which can be processed by processing machines for automating execution of the administrative process [apply list is submitted 138 for execution via a managed entity configuration management module 140; p. 4, paragraph 0053]. Bolder does not specifically teach creating a generic command framework to capture the structure of all types of administrative commands used in the administrative processes.

However, Murray teaches creating a generic command framework [CLI framework 220 is coded in a generic fashion and is adapted to load at run-time grammar files 226; col. 10, lines 28 – 41] to capture the structure of all types of administrative commands used in the administrative processes [CLI dictionary 530 further codifies the CLI commands, the relations between CLI commands, and how the commands map to managed data network entities 510; col. 10, line 61 – col. 11, line 3].

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to modify the invention of Bolder to incorporate the features of Murray. One of ordinary skill in the art would have been motivated to make the combination because this automates the Command Line Interface (CLI) command entry task in a network management and service provisioning environment comprised of

diverse data network entities each of which has an associated CLI vocabulary [col. 12, line 65 – col. 13, line 8 of Murray].

11. As to claim 4, Bolder as modified teaches a method for automating execution of administrative processes in a local or a networked environment, the administrative processes comprising at least one administrative task and the administrative task comprising administrative commands [col. 12, line 65 – col. 13, line 8 of Murray], the method comprising the steps of:

a. creating a generic command framework [col. 10, lines 28 – 41 of Murray] that can be used to capture structures of all types of administrative commands used in the administrative processes [col. 10, line 61 – col. 11, line 3 of Murray];

b. generating profiles [p. 3, paragraph 0037 of Bolder] by capturing the administrative commands in the generic command framework in a structured mark up language to create profiles [grammar files may be written in eXtensible Markup Language; col. 10, lines 40 – 47 of Murray], the profiles being collections of administrative commands that define an administrative task [p. 4, paragraph 0059 of Bolder];

c. generating network maps, each network map containing details of one or more servers in the networked environment [p. 3, paragraph 0050 of Bolder];

d. generating administrative lists, each administrative list being a collection of one or more profiles and zero or more network maps, that define administrative tasks,

Art Unit: 2194

which are required to be performed for executing an administrative process [pp. 3 – 4, paragraph 0051 of Bolder];

e. selecting a first set of administrative lists based on a selected administrative process [selecting script templates 122 for inclusion in an apply list; pp. 3 – 4, paragraph 0051 of Bolder], which is to be executed;

f. selecting the first profile from among the profiles contained in the selected first set of administrative lists [requests and retrieves 142 target managed entity parameters 212 from the NMS DB 210 associated with the NMS 200 based on NMS ParameterIDs specified and populates 144 each script template 122 in the apply list to generate a custom CLI command script for each specific target managed entity 232; p. 4, paragraph 0059 of Bolder];

g. processing the selected profile [p. 4, paragraph 0059 of Bolder], the processing of selected profile comprising the steps of:

i. parsing the selected profile to generate events [grammar files 226 are interpreted by a parser 222; col. 10, lines 40 – 47 of Murray];

ii. executing the generated events using administered components [event driven as a current state of the managed data transport network; col. 7, lines 8 – line 15 of Murray], the administered components comprising at least one of local system devices, networked system devices, software programs and system commands, which are used to execute the generated events [col. 11, lines 30 – 35 of Murray];

iii. collecting messages generated by administered components in response to the execution of generated events [changes the data network state and/or provides an update of the data network state; col. 8, lines 32 – 38 of Murray];

iv. updating the selected profile based on the collected messages [CLlupdate actions; col. 9, lines 25 – 35 of Murray];

v. deciding if the updated profile needs to be processed again, the decision being made based on the messages using which the selected profile has been updated [p. 4, paragraph 0056 of Bolder];

vi. if the updated profile needs to be processed again, then performing the steps of

a) selecting the updated profile for processing [col. 9, lines 25 – 35 of Murray];

b) repeating step f for the selected profile [p. 4, paragraph 0059 of Bolder];

else performing the steps of

c) selecting next profile to be processed from the selected first set of administrative lists based on the administrative tasks to be performed [col. 11, lines 33 – 45 of Murray]; and

d) repeating step f for the selected new profile [p. 4, paragraph 0059 of Bolder];
wherein the execution of administrative processes is automated by the automated processing of profiles that define the administrative tasks, which are required to be performed for executing the administrative processes [p. 4, paragraph 0053 of Bolder].

12. As to claim 10, Bolder as modified teaches a system for automating administrative processes in a local or a networked environment [col. 12, line 65 – col. 13, line 8 of Murray], the administrative processes comprising administrative tasks and the administrative tasks comprising administrative commands [p. 4, paragraph 0059 of Bolder], the system comprising:

a. a profile generator for generating profiles [p. 3, paragraph 0037 of Bolder] in a structured mark up language [col. 10, lines 40 – 47 of Murray], the profiles being collections of administrative commands that define an administrative task [p. 4, paragraph 0059 of Bolder];

b. a network map generator for generating network maps, the network maps containing details of one or more servers in the networked environment [p. 3, paragraph 0050 of Bolder];

c. an administrative list generator for generating administrative lists, each administrative list being a collection of one or more profiles that define administrative tasks, which are required to be performed for executing an administrative process [pp. 3 – 4, paragraph 0051 of Bolder];

d. a front-end for enabling at least one function from among the functions of selecting profiles contained in administrative lists, reading the selected profiles and transforming the selected profiles [pp. 3 – 4, paragraph 0045 - 0051 of Bolder]; and

e. a back-end comprising:

i. an event-driven machine for processing the selected profiles written in a structured mark up language [col. 7, lines 8 – line 15 of Murray]; and

ii. one or more controllers for connecting the front-end with the event-driven machine for processing the selected profiles, the controllers passing the selected profiles from the front-end to the event-driven machine [col. 11, lines 30 – 35 of Murray].

13. As to claim 15, Bolder as modified teaches a computer program product, comprising a computer usable medium having a computer readable program code embodied therein, for capturing and automating the execution of administrative processes in a local or a networked environment [col. 10, line 61 – col. 11, line 3 of Bolder], the administrative processes comprising administrative tasks and the administrative tasks comprising administrative commands [p. 3, paragraph 0041 of Bolder], the computer program product comprising:

a. program instruction means for capturing the administrative commands [col. 10, line 61 – col. 11, line 3 of Murray] in a generic command framework [col. 10, lines 28 – 41 of Murray] in a structured mark up language to create profiles [col. 10, lines 40 – 47 of Murray], the profiles being collections of administrative commands that define an administrative task [p. 4, paragraph 0059 of Bolder];

b. program instruction means for generating network maps, each network map containing details of one or more servers in the networked environment [p. 3, paragraph 0050 of Bolder];

c. program instruction means for generating administrative lists, each administrative list being a collection of one or more profiles that define administrative

tasks, which are required to be performed for executing an administrative process [pp. 3 – 4, paragraph 0051 of Bolder];

d. program instruction means for selecting a first set of administrative lists based on a selected administrative process, which is to be executed [pp. 3 – 4, paragraph 0051 of Bolder];

e. program instruction means for selecting the first profile from among the profiles contained in the selected first set of administrative lists [p. 4, paragraph 0059 of Bolder];

f. program instruction means for processing the selected profile, the program instruction means for processing the selected profile comprising [p. 4, paragraph 0059 of Bolder];

i. program instruction means for parsing the selected profile to generate events [col. 10, lines 40 – 47 of Murray];

ii. program instruction means for executing the generated events using administered components [col. 7, lines 8 – line 15 of Murray], the administered components comprising at least one of local system devices, networked system devices, software programs and system commands, which are used to execute the generated events [col. 11, lines 30 – 35 of Murray];

iii. program instruction means for collecting messages generated by administered components after the execution of generated events [col. 8, lines 32 – 38 of Murray];

iv. program instruction means for updating the selected profile based on the collected messages [CLUpdate actions; col. 9, lines 25 – 35 of Murray];

v. program instruction means for checking if the updated profile needs to be processed again [p. 4, paragraph 0056 of Bolder]; and

vi. program instruction means for selecting the next profile contained in the selected first set of administrative lists [col. 9, lines 25 – 35 of Murray].

14. As to claim 19, Bolder as modified teaches a system for automating administrative processes in a local or a networked environment [col. 12, line 65 – col. 13, line 8 of Murray], the administrative processes comprising administrative tasks and the administrative tasks comprising administrative commands [p. 4, paragraph 0059 of Bolder], the system comprising:

a. a profile generator for generating profiles [p. 3, paragraph 0037 of Bolder] in a structured mark up language [col. 10, lines 40 – 47 of Murray], the profiles being collections of administrative commands that define an administrative task [p. 4, paragraph 0059 of Bolder];

b. a network map generator for generating network maps, the network maps containing details of one or more servers in the networked environment [p. 3, paragraph 0050 of Bolder];

c. an administrative list generator for generating administrative lists, each administrative list being a collection of one or more profiles that define administrative tasks, which are required to be performed for executing an administrative process [pp. 3 – 4, paragraph 0051 of Bolder];

d. a front-end for enabling at least one function from among the functions of selecting a profile contained in administrative lists, reading the selected profile and transforming the selected profiles [pp. 3 – 4, paragraph 0045 - 0051 of Bolder]; and

e. a back-end comprising:

i. an event-driven machine for processing the selected profile written in a structured mark up language [col. 7, lines 8 – line 15 of Murray], the event-driven machine further comprising:

a) one or more generators, which generate events by parsing the selected profile [col. 10, lines 40 – 47 of Murray];

b) one or more filters, the filters directing the generated events to administered components and causing execution of the generated events at the administered components, thereby leading to the generation of messages by the administered devices in response to the execution of the generated events [col. 11, lines 3 – 12 of Murray]; and

c) one or more handlers, for updating the selected profile with the messages generated by the administered components [col. 11, lines 33 – 45 of Murray];

ii. one or more controllers for connecting the front-end with the event-driven machine for processing the selected profile, the controllers transforming the profile from one format of structured mark up language to another format, if required, based on a set of pre-defined rules and passing the selected profile from the front-end to the event-driven machine [col. 11, lines 30 – 35 of Murray].

15. As to claim 2, Bolder as modified teaches the step of generating administrative lists comprises defining associations between profiles present in the administrative list and one or more network maps, the association between a profile and one or more network maps defining the server(s) on which the administrative task defined by the profile needs to be performed [p. 4, paragraph 0059 of Bolder].

16. As to claim 3, Bolder as modified teaches wherein the administrative commands are captured in a machine-readable format by the step of writing them according to the generic command framework in a structured mark up language [col. 10, lines 28 – 41 of Murray].

17. As to claim 5, Bolder as modified teaches a. determining if the selected profile needs to be transformed before processing, the transformation being required when the selected profile needs to be converted from a first format of structured mark up language to a second format of the structured mark up language [col. 10, lines 28 – 45 of Bolder]; if the selected profile needs to be transformed, then b. transforming the selected profile from the first format to the second format using a set of pre-defined rules [col. 10, lines 28 – 45 of Bolder]; else c. processing the selected profile in the first format [col. 11, lines 30 – 35 of Murray].

18. As to claim 6, Bolder as modified teaches an administrative list further comprises associations between the profiles present in the administrative list and one or more

network maps [p. 3, paragraph 0050 of Bolder], the association between a profile and one or more network maps defining the server(s) in the networked environment, on which the administrative task defined by the profile in the administrative list needs to be performed [p. 4, paragraph 0059 of Bolder].

19. As to claim 7, Bolder as modified teaches a. directing the generated events to administered components using which the generated events can be executed [col. 7, lines 8 – line 15 of Murray]; and b. executing the generated events using the administered components to which the generated events have been directed [col. 11, lines 30 – 35 of Murray].

20. As to claim 8, Bolder as modified teaches the generated messages comprise control messages and system messages [col. 10, lines 15 – 25 of Murray].

21. As to claim 9, Bolder as modified teaches the step of updating the selected profile based on the collected messages comprises the steps of: a. appending the generated messages in a structured mark up language to the selected profile; and b. storing the updated profile [col. 9, lines 25 – 35 of Murray].

22. As to claim 11, Bolder as modified teaches the event-driven machine for processing the selected profiles comprises: a. one or more generators, which generate events by parsing the selected profile [col. 10, lines 40 – 47 of Murray]; b. one or more

filters, the filters directing the generated events to administered components and causing execution of the generated events at the administered components, thereby leading to the generation of messages by the administered devices in response to the execution of the generated events [col. 11, lines 3 – 12 of Murray]; and c. one or more handlers, for updating the selected profile with messages generated by the administered components [col. 11, lines 33 – 45 of Murray].

23. As to claim 12, Bolder as modified teaches the administered components comprise at least one of local devices, networked devices, software programs and system commands, which are used for executing the generated events [col. 11, lines 30 – 35 of Murray].

24. As to claim 13, Bolder as modified teaches the controller further transforms the selected profile from a first format of structured mark up language to a second format of the structured mark up language, based on a set of pre-defined rules [pp. 3 – 4, paragraph 0045 - 0051 of Bolder].

25. As to claim 14, Bolder as modified teaches the pre-defined rules are stored locally in the controller where the transformation of selected profiles is performed, or are stored outside the controller in the networked environment [col. 11, lines 30 – 35 of Murray].

26. As to claim 16, Bolder as modified teaches a program instruction means for transforming the selected profile before processing, the transformation being required when the selected profile needs to be converted from a first format of structured mark up language to a second format of the structured mark up language [col. 10, lines 28 – 45 of Bolder].

27. As to claim 17, Bolder as modified teaches the program instruction means for executing the generated events further comprises: a. program instruction means for directing the generated events to administered components using which the generated events can be executed [col. 11, lines 3 – 12 of Murray]; and b. program instruction means for causing execution of the generated events using the administered components to which the generated events have been directed [col. 13, line 65 – col. 14, line 8 of Murray].

28. As to claim 18, Bolder as modified teaches wherein the program instruction means for updating a profile based on the collected messages comprises: a. program instruction means for appending the generated messages in a structured mark up language to the selected profile; and b. program instruction means for storing the updated profile [col. 9, lines 25 – 35 of Murray].

29. As to claim 20, Bolder as modified teaches the administered components comprise at least one of local devices, networked devices, software programs and

Art Unit: 2194

system commands, which are used for executing the generated events [col. 11, lines 30 – 35 of Murray].

CONTACT INFORMATION

30. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571)272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Li B. Zhen
Primary Examiner
Art Unit 2194

/Li B. Zhen/
Primary Examiner, Art Unit 2194